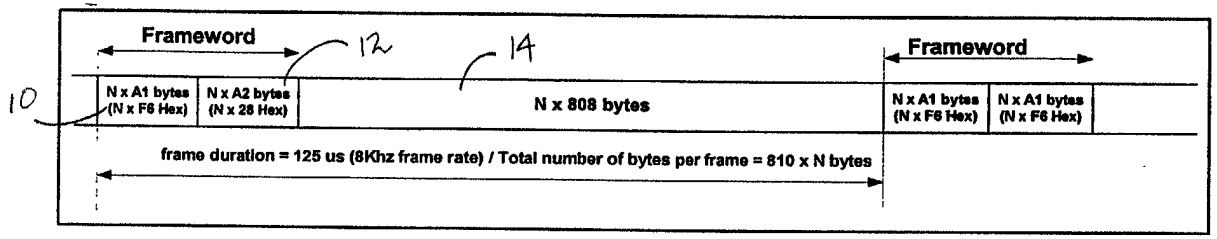
Frameword
12
14
Frameword
10
N x A1 bytes (N x F6 Hex)
N x A2 bytes (N x 28 Hex)
N x 808 bytes
N x A1 bytes (N x F6 Hex)
N x A1 bytes (N x F6 Hex)
frame duration = 125 us (8Khz frame rate) / Total number of bytes per frame = 810 x N bytes

FIG. 1

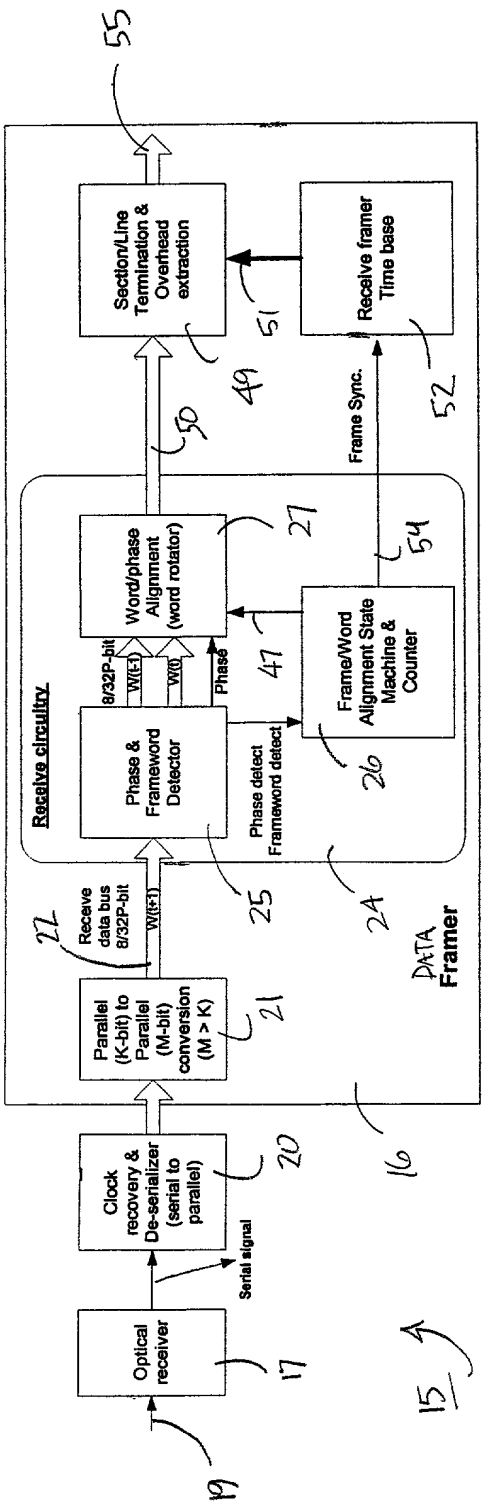
Optical receiver
Clock recovery & De-serializer (serial to parallel)
Serial signal
Parallel (K-bit) to Parallel (M-bit) conversion (M > K)
Receive data bus 8/32P-bit
W(i+1)
Receive circuitry
Phase & Frameword Detector
8/32P-bit
W(i+1)
W(i)
Phase
Phase detect Frameword detect
Word/phase Alignment (word rotator)
Frame/Word Alignment State Machine & Counter
Frame Sync.
Section/Line Termination & Overhead extraction
Receive framer Time base
DATA Framer
19
17
15
20
21
22
25
24
16
26
47
27
54
50
49
51
52
55

FIG. 2

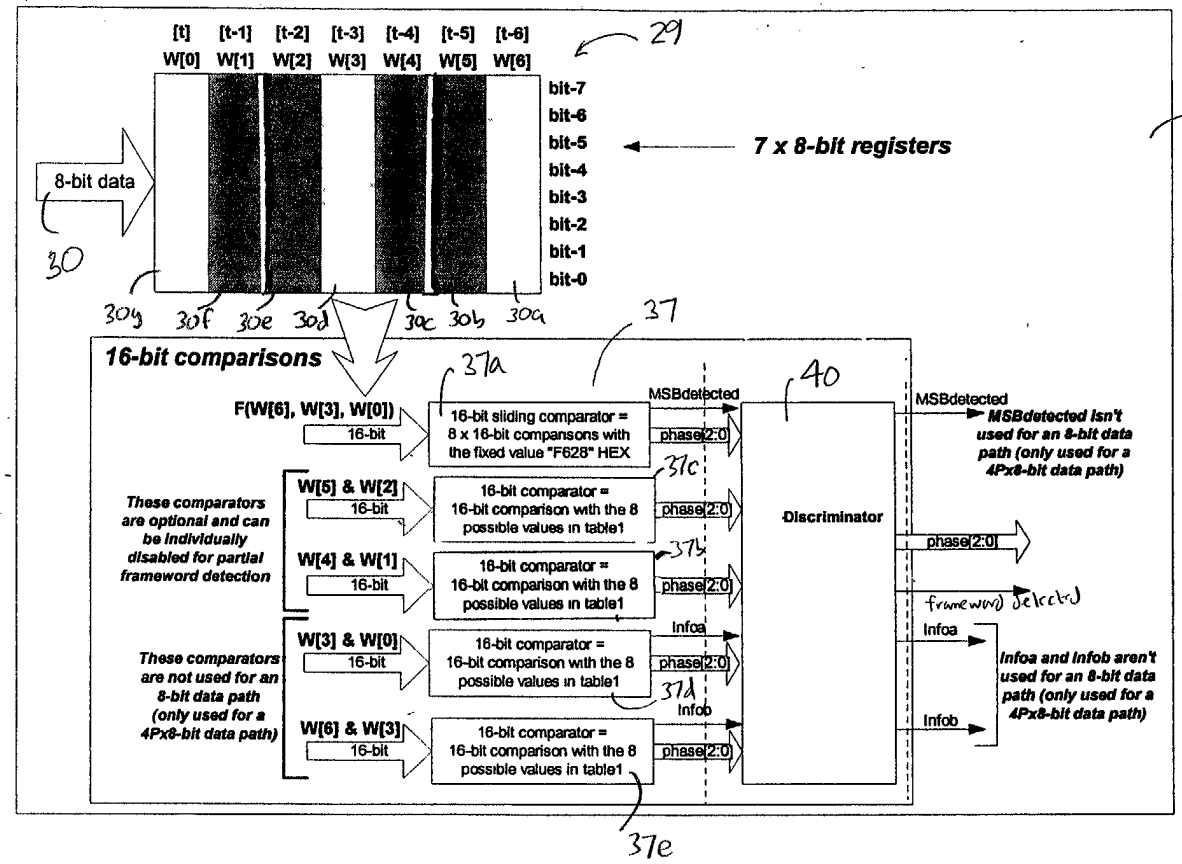
[t] [t-1] [t-2] [t-3] [t-4] [t-5] [t-6]
W[0] W[1] W[2] W[3] W[4] W[5] W[6]
29
bit-7
bit-6
bit-5
bit-4
bit-3
bit-2
bit-1
bit-0
7 x 8-bit registers
25
8-bit data
30
30g 30f 30e 30d 30c 30b 30a
37
16-bit comparisons
37a
F(W[6], W[3], W[0])
16-bit
16-bit sliding comparator = 8 x 16-bit compansons with the fixed value "F628" HEX
MSBdetected
phase[2:0]
40
Discriminator
MSBdetected
MSBdetected isn't used for an 8-bit data path (only used for a 4Px8-bit data path)
These comparators are optional and can be individually disabled for partial frameword detection
W[5] & W[2]
16-bit
16-bit comparator = 16-bit comparison with the 8 possible values in table1
37c
W[4] & W[1]
16-bit
16-bit comparator = 16-bit comparison with the 8 possible values in table1
37b
phase[2:0]
frameword detected
These comparators are not used for an 8-bit data path (only used for a 4Px8-bit data path)
W[3] & W[0]
16-bit
16-bit comparator = 16-bit comparison with the 8 possible values in table1
Infoa
37d
Infoa
Infoa and Infob aren't used for an 8-bit data path (only used for a 4Px8-bit data path)
W[6] & W[3]
16-bit
16-bit comparator = 16-bit comparison with the 8 possible values in table1
Infob
Infob
37e

FIG. 3

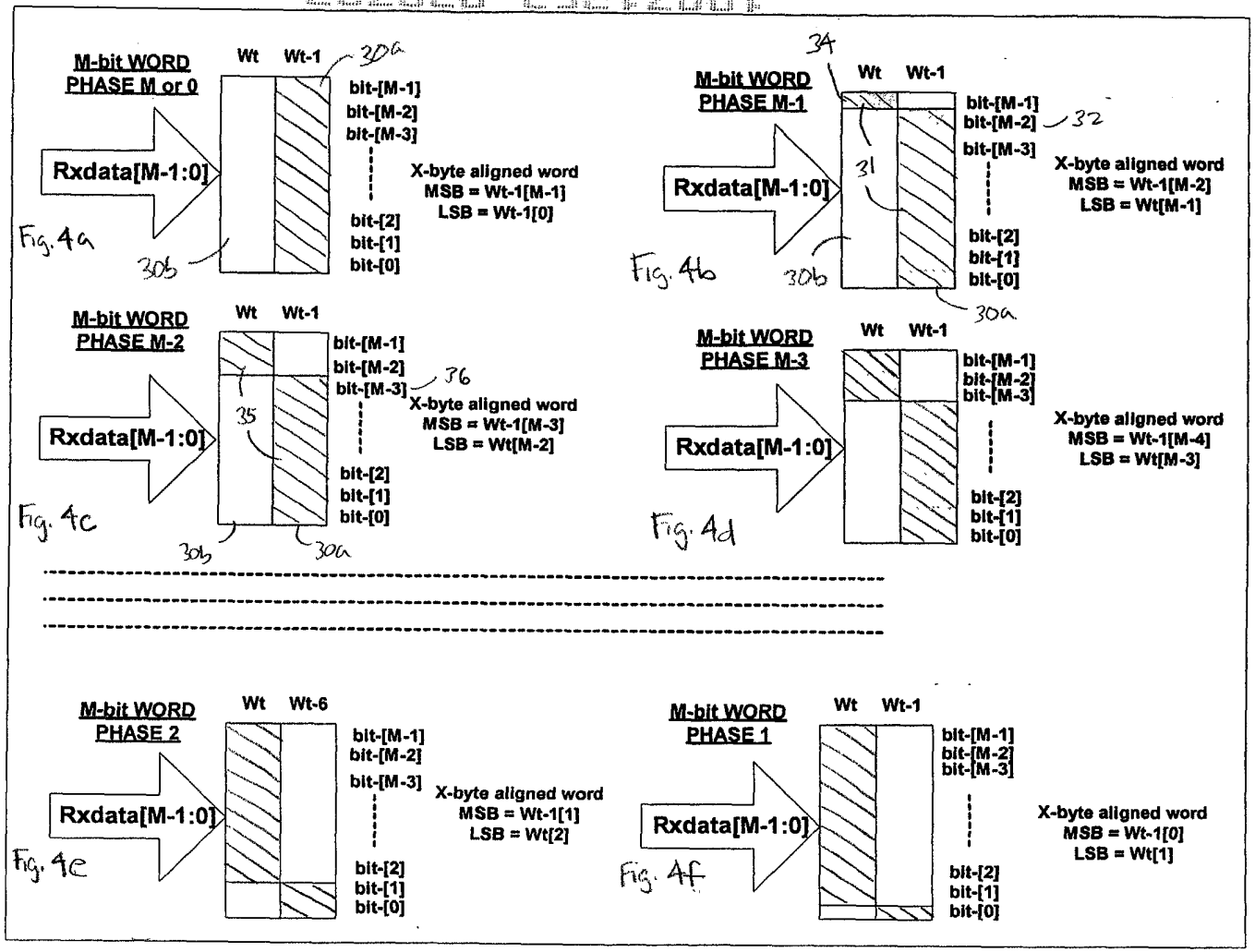
M-bit WORD
PHASE M or 0
Wt
Wt-1
30a
bit-[M-1]
bit-[M-2]
bit-[M-3]
bit-[2]
bit-[1]
bit-[0]
X-byte aligned word
MSB = Wt-1[M-1]
LSB = Wt-1[0]
Rxdata[M-1:0]
Fig. 4a
30b
34
M-bit WORD
PHASE M-1
32
31
X-byte aligned word
MSB = Wt-1[M-2]
LSB = Wt[M-1]
Fig. 4b
M-bit WORD
PHASE M-2
36
35
X-byte aligned word
MSB = Wt-1[M-3]
LSB = Wt[M-2]
Fig. 4c
M-bit WORD
PHASE M-3
X-byte aligned word
MSB = Wt-1[M-4]
LSB = Wt[M-3]
Fig. 4d
M-bit WORD
PHASE 2
Wt-6
X-byte aligned word
MSB = Wt-1[1]
LSB = Wt[2]
Fig. 4e
M-bit WORD
PHASE 1
X-byte aligned word
MSB = Wt-1[0]
LSB = Wt[1]
Fig. 4f

FIG. 4

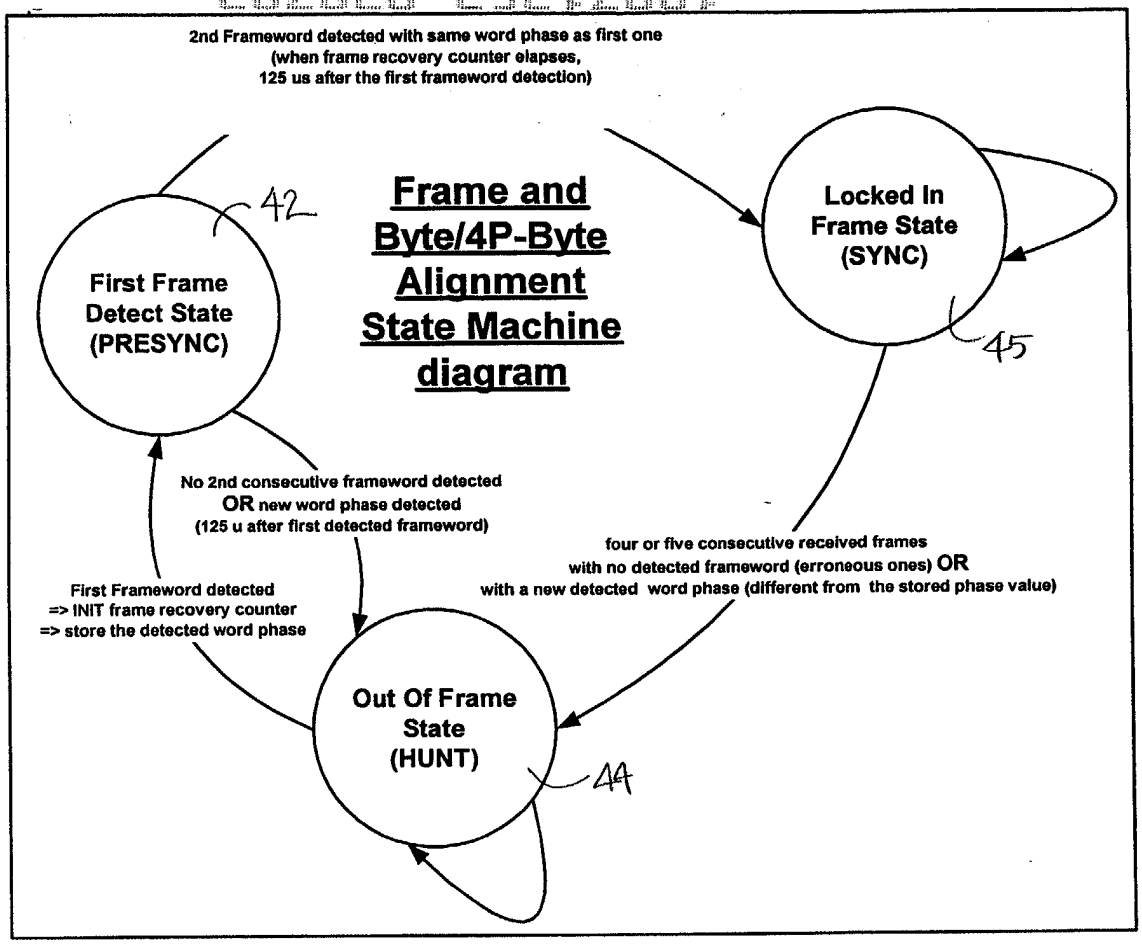
2nd Frameword detected with same word phase as first one
(when frame recovery counter elapses,
125 us after the first frameword detection)
Frame and Byte/4P-Byte Alignment State Machine diagram
First Frame Detect State (PRESYNC)
42
Locked In Frame State (SYNC)
45
Out Of Frame State (HUNT)
44
No 2nd consecutive frameword detected
OR new word phase detected
(125 u after first detected frameword)
First Frameword detected
=> INIT frame recovery counter
=> store the detected word phase
four or five consecutive received frames
with no detected frameword (erroneous ones) OR
with a new detected word phase (different from the stored phase value)

FIG. 5

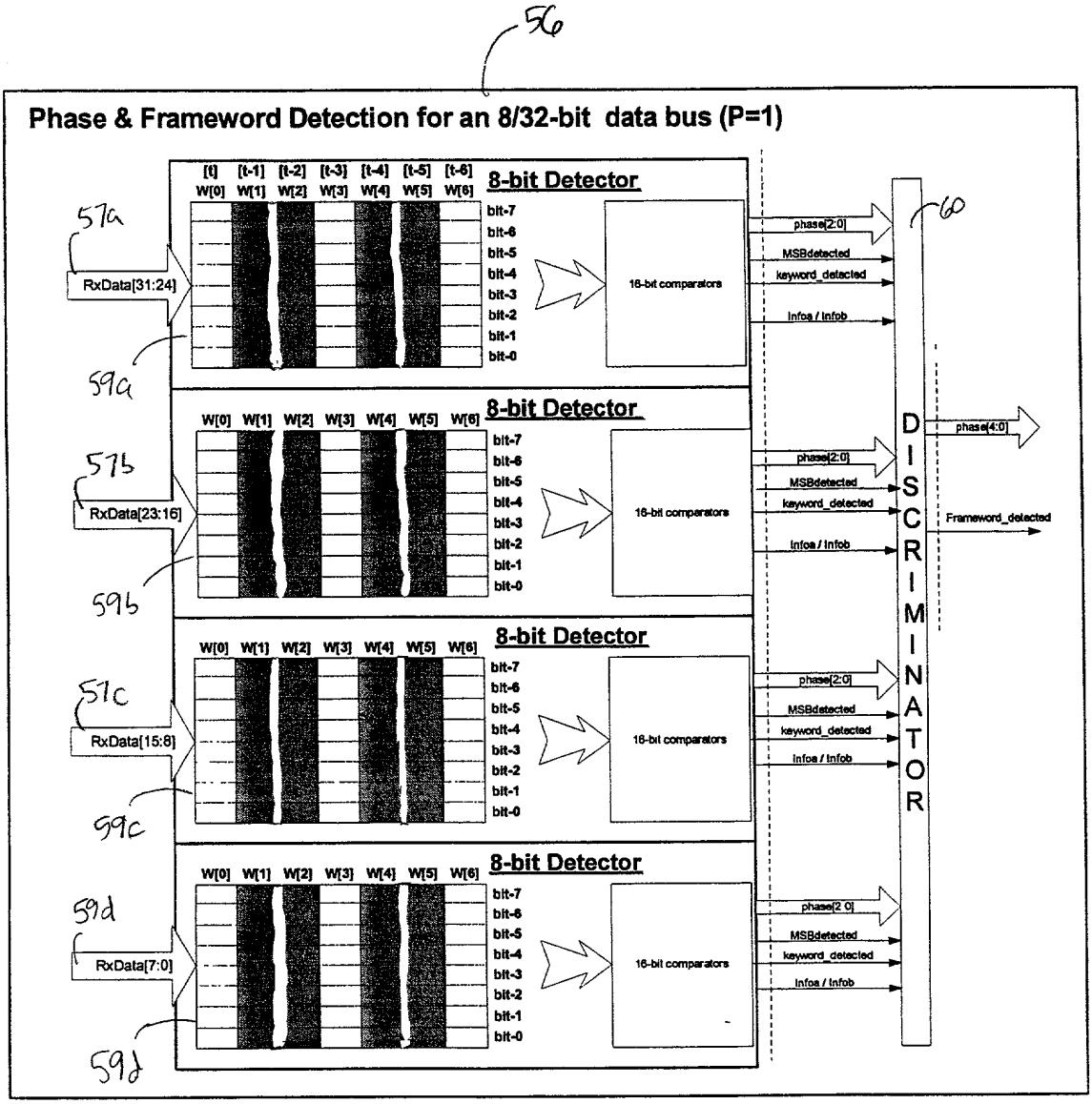
56
Phase & Frameword Detection for an 8/32-bit data bus (P=1)
8-bit Detector
[t] [t-1] [t-2] [t-3] [t-4] [t-5] [t-6]
W[0] W[1] W[2] W[3] W[4] W[5] W[6]
bit-7
bit-6
bit-5
bit-4
bit-3
bit-2
bit-1
bit-0
57a
RxData[31:24]
59a
16-bit comparators
phase[2:0]
MSBdetected
keyword_detected
Infoa / Infob
57b
RxData[23:16]
59b
57c
RxData[15:8]
59c
57d
RxData[7:0]
59d
60
DISCRIMINATOR
phase[4:0]
Frameword_detected

FIG. 6

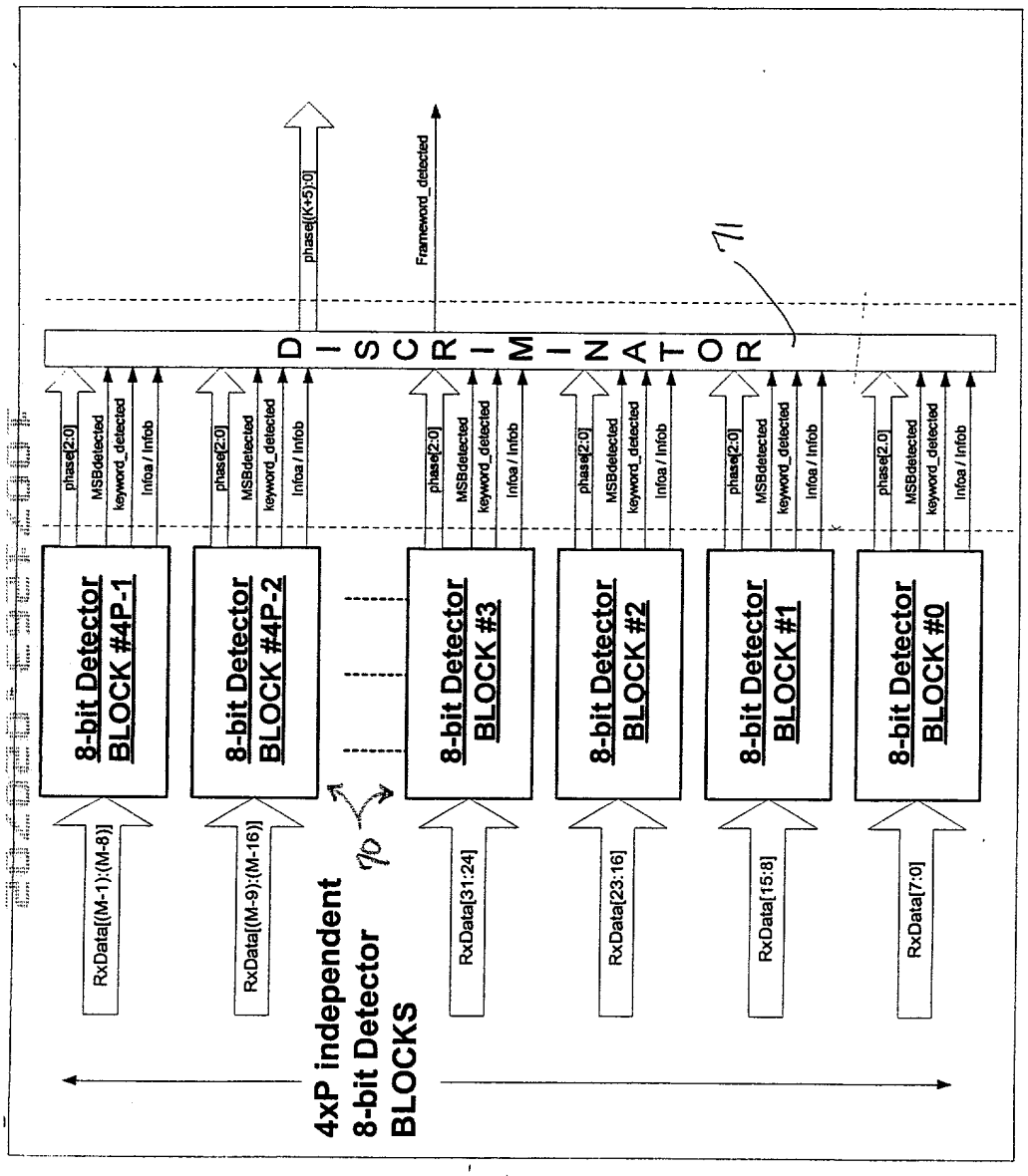
DISCRIMINATOR
phase[(K+5):0]
Frameword_detected
71
phase[2:0]
MSBdetected
keyword_detected
Infoa / Infob
8-bit Detector BLOCK #4P-1
8-bit Detector BLOCK #4P-2
8-bit Detector BLOCK #3
8-bit Detector BLOCK #2
8-bit Detector BLOCK #1
8-bit Detector BLOCK #0
RxData[(M-1):(M-8)]
RxData[(M-9):(M-16)]
RxData[31:24]
RxData[23:16]
RxData[15:8]
RxData[7:0]
4xP independent 8-bit Detector BLOCKS

FIG. 7